

Maven | M2E | Tycho

by Sandip Chitale

Ant vs Maven

- Ant
 - Procedural
 - How to do it
- Maven
 - Declarative
 - Get this done
 - Convention over configuration
 - Standard project structure
 - “Magic” hidden in plugins and goals

Simple Ant build file

```
<project name="antproject" default="dist" basedir=".">
  <description>
    simple example build file
  </description>
  <!-- set global properties for this build -->
  <property name="src" location="src"/>
  <property name="build" location="build"/>
  <property name="dist" location="dist"/>

  <target name="init">
    <!-- Create the time stamp -->
    <tstamp/>
    <!-- Create the build directory structure used by compile -->
    <mkdir dir="${build}"/>
  </target>

  <target name="compile" depends="init"
    description="compile the source " >
    <!-- Compile the java code from ${src} into ${build} -->
    <javac srcdir="${src}" destdir="${build}"/>
  </target>

  <target name="dist" depends="compile"
    description="generate the distribution" >
    <!-- Create the distribution directory -->
    <mkdir dir="${dist}/lib"/>

    <!-- Put everything in ${build} into the MyProject-${DSTAMP}.jar file -->
    <jar jarfile="${dist}/lib/MyProject-${DSTAMP}.jar" basedir="${build}"/>
  </target>

  <target name="clean"
    description="clean up" >
    <!-- Delete the ${build} and ${dist} directory trees -->
    <delete dir="${build}"/>
    <delete dir="${dist}"/>
  </target>
</project>
```

Simple POM

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.emc.xcp</groupId>
  <artifactId>mavenproject</artifactId>
  <version>1.0</version>
  <packaging>jar</packaging><!-- convention over configuration -->
</project>
```

Concepts

- POM
- Artifact Repositories
 - Artifacts
 - Plugins
 - Archetypes
- Build
 - Reactor
 - Profiles
 - Activate and deactivate to vary builds

POM

- GAV
- POM Relationships
 - Dependencies
 - Scopes – will talk about it later
 - Parent/Child
 - Inheritance
 - Super POM
 - Properties and configurations are inherited
 - Notion of effective POM
 - Aggregator/Module

GAV

- Group Id
- Artifact Id
- Version
 - Snapshot (development mode)
 - Release
 - `major.minor.service-qualifier`
- Packaging
- Classifier

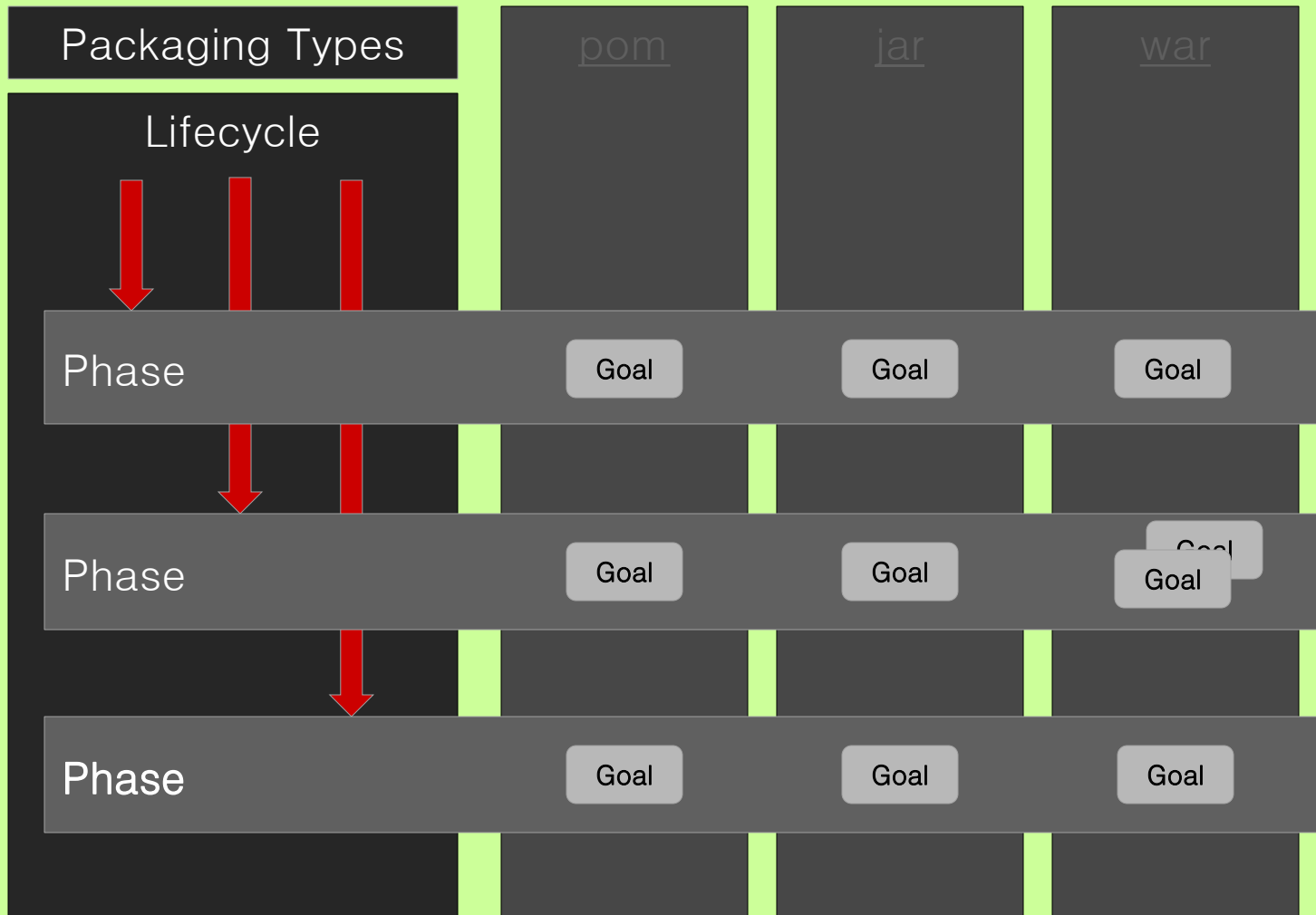
POM Relationships

- Dependencies
 - Scopes
 - Exclusion
- Parent/Child
 - Inheritance
 - When things need to be shared
- Aggregator/Module
 - Orchestrate the build
 - When things need to be build together
- Parent/Child vs Aggregator/Module

POM

- Packaging Type e.g. pom, jar, war, ear
- Build
 - Lifecycle (controlled by packaging type)
 - Phases
 - Goals attached to phases
 - Contributed by **Plugins**
 - Standard Lifecycles
 - Clean
 - Default
 - Site

Lifecycle (visualization)



Standard Lifecycles – default

Lifecycle	Phases	Packaging		
		pom	maven-plugin plugin*:goal	jar
Default				
	validate			
	initialize			
	generate-sources			
	process-sources			
	generate-resources		plugin:descriptor	
	process-resources		resources:resources	resources:resources
	compile		compiler:compile	compiler:compile
	process-classes			
	generate-test-sources			
	process-test-sources			
	generate-test-resources			
	process-test-resources		resources:testResources	resources:testResources
	test-compile		compiler:testCompile	compiler:testCompile
	process-test-classes			
	test		surefire:test	surefire:test
	prepare-package			
to target folder	package	site:attach-descriptor	jar:jar plugin:addPluginArtifactMetadata	jar:jar
	pre-integration-test			
	integration-test			
	post-integration-test			
	verify			
to .m2/ Repository	install	install:install	install:install plugin:updateRegistry	install:install
to group repository	deploy	deploy:deploy		deploy:deploy

* maven-pluginname-plugin – convention over configuration

Standard Lifecycles – clean

Lifecycle	Phases	Packaging		
		pom	maven-plugin	jar
		plugin:goal		
Clean				
	pre-clean			
	clean			clean:clean
	post-clean			

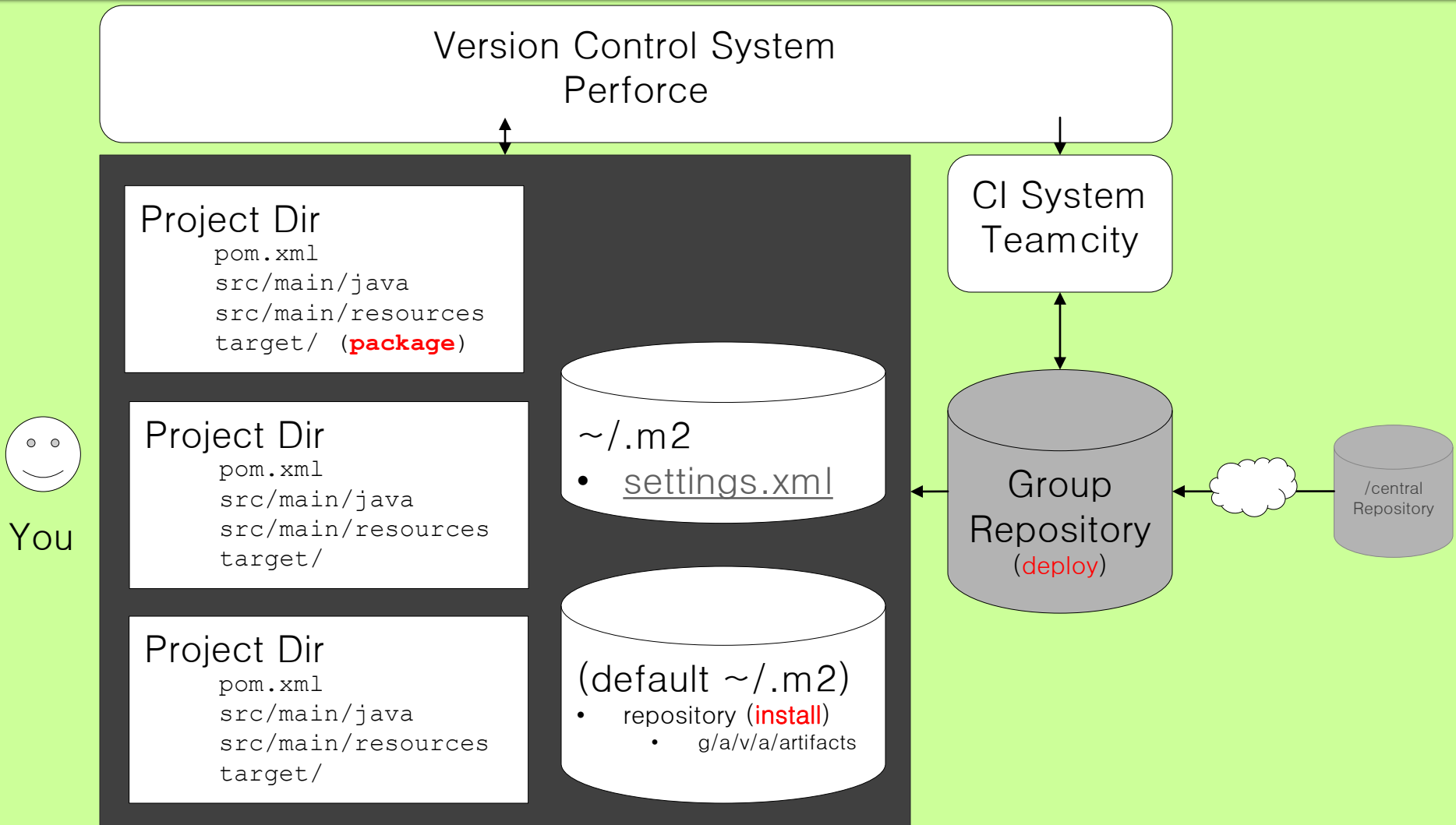
Standard Lifecycles – site

Lifecycle	Phases	Packaging		
		pom	maven-plugin plugin:goal	jar
Site				
	pre-site			
	site	site:site	site:site	site:site
	post-site			
	site-deploy	site:deploy	site:deploy	site:deploy

Dependencies

- GAV
 - Versions
 - SNAPSHOT
 - Release
 - maven-release-plugin
 - Version Ranges [(,)]
 - Scope
 - Compile
 - Test
 - Provided
 - Runtime
 - System (not portable/not recommended)

Architecture



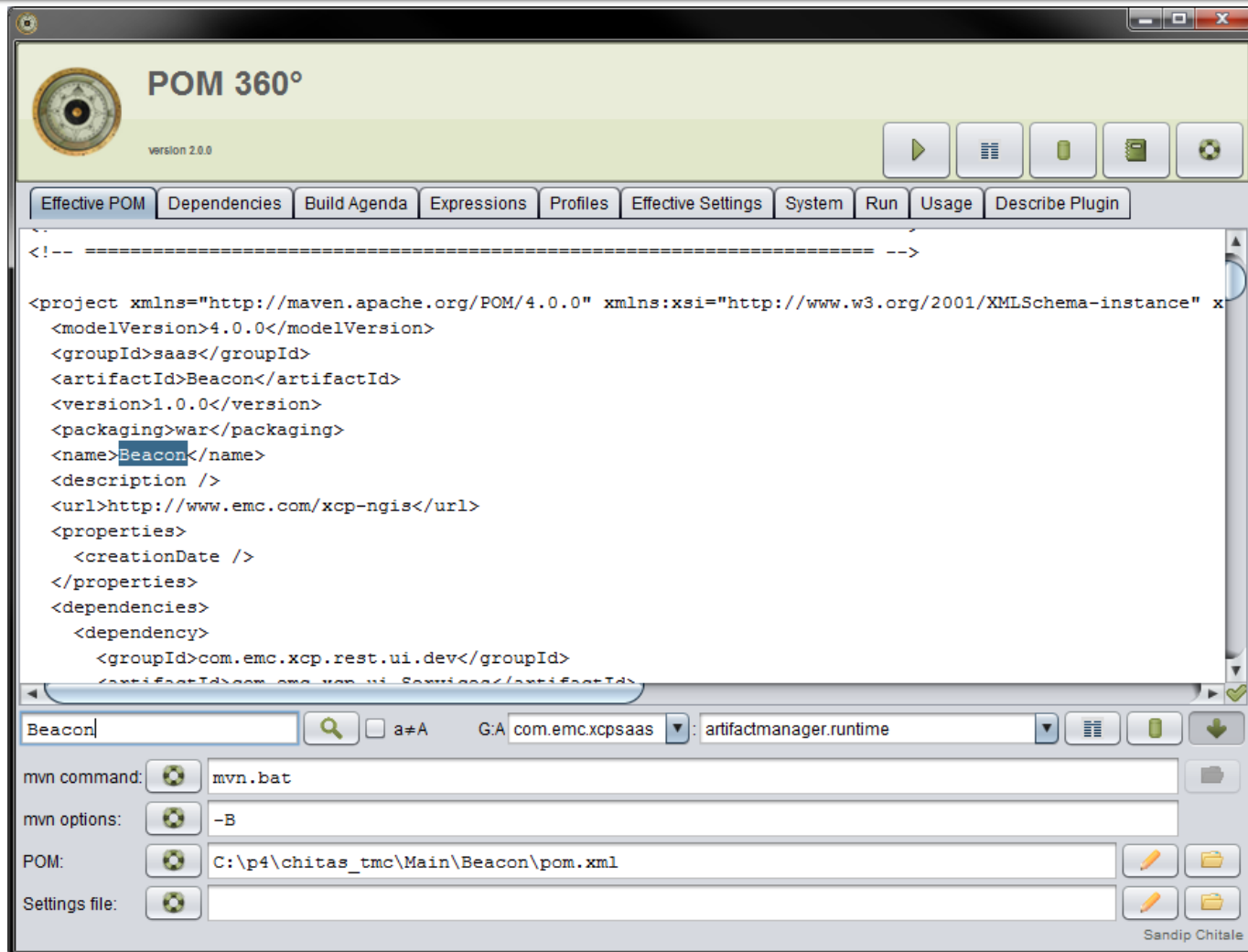
Settings

- Location of repositories
 - Remote
 - Cache
- Profiles

Archetypes

- Standardize Project Structures
- Templating mechanism

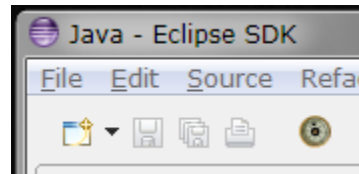
POM360



POM 360

- Makes heavy use of
 - maven-help-plugin
 - maven-dependency-plugin

POM 360 Eclipse integration



M2E

- Embedded Maven+Eclipse integraion
- Maven POM is the boss
 - pom.xml -> .project and .classpath
- Ties into Eclipse incremental Build
- Launch Configurations for maven builds
- Structured pom.xml editor
- New Maven Project Wizard
- Import Maven Project from Team Repo support

Tycho

- Maven based Eclipse builds to build
 - Plugins|Features|Update Sites| Products
- Replaces clucky P2 builds
- Eclipse is the boss
 - .project and .classpath -> pom.xml
- Packaging types
 - Plugins|Features|Update Sites| Products
- Tycho plugin goals to build

Reference

Maven Reference (download)

M2E

Tycho

POM360

Q&A

Thanks!